

we claim

We CLAIMS:

1 1. A method of accessing values stored in a cache used
2 by a processor of a computer system, comprising the steps
3 of:

4 loading a plurality of memory blocks from a memory
5 device connected to the processor via a system bus into
6 respective cache lines of the cache;

7 writing address tags associated with the memory blocks
8 into first and second cache directories of the cache; and

9 reading a first memory block from the cache using the
10 first cache directory, while reading a second memory block
11 from the cache using the second cache directory.

1 2. The method of Claim 1 wherein the first and second
2 cache directories are redundant, and said writing step
3 writes a given one of the address tags to a specific line of
4 the first directory and to a specific line of the second
5 directory that corresponds to the specific line of the first
6 directory.

1 3. The method of Claim 1 wherein the cache has a single
2 cache entry array, and said step of reading the first memory
3 block while reading the second memory block includes the
4 steps of:

5 constructing a first control signal for the first
6 memory block based on a first location
7 in the first directory of an address tag associated with the
8 first memory block;

26

9 applying the first control signal to a first
10 multiplexer having inputs connected to the cache entry
11 array;

12 constructing a second control signal for the second
13 memory block based on a second location in the second
14 directory of an address tag associated with the second
15 memory block; and

16 applying the second control signal to a second
17 multiplexer having inputs connected to the cache entry
18 array.

1 4. The method of Claim 1 wherein the cache has first
2 and second cache entry arrays, and said step reading the
3 first memory block while reading the second memory block
4 includes the steps of:

5 constructing a first control signal for the first
6 memory block based on a first location in the first
7 directory of an address tag associated with the first memory
8 block;

9 applying the first control signal to a first
10 multiplexer having inputs connected to the first cache entry
11 array;

12 constructing a second control signal for the second
13 memory block based on a second location in the second
14 directory of an address tag associated with the second
15 memory block; and

16 applying the second control signal to a second
17 multiplexer having inputs connected to the second cache
18 entry array.

27

1 5. The method of Claim 1 wherein the first and second
2 memory blocks are read in a single clock cycle of the
3 processor.

1 6. The method of Claim 2 wherein each of the first and
2 *bad* second cache directories have a plurality of congruence
3 classes each having a plurality of lines for storing the
4 address tags, and said step of the processor reading the
5 first memory block while the system bus is reading the
6 second memory block includes the steps of:

7 associating a first requested address with a first
8 congruence class in the first cache directory;

9 comparing each of the address tags stored in the first
10 congruence class with a portion of the first requested
11 address;

12 associating a second requested address with a second
13 congruence class in the second cache directory; and

14 comparing each of the address tags stored in the second
15 congruence class with a portion of the second requested
16 address.

1 7. The method of Claim 3 wherein the first cache
2 directory is connected to a first interconnect on a
3 processor side of the cache, and the second cache directory
4 is connected to a second interconnect on a system bus side
5 of the cache, and said step of reading the first memory
6 block while reading the second memory block further includes
7 the steps of:

28

8 presenting the first memory block to the first
9 interconnect by connecting the first interconnect to an
10 output of the first multiplexer; and

11 presenting the second memory block to the second
12 interconnect by connecting the second interconnect to an
13 output of the second multiplexer.

1 7

2 8. The method of Claim 4 wherein the first cache
3 directory is connected to a first interconnect on a
4 processor side of the cache, and the second cache directory
5 is connected to a second interconnect on a system bus side
6 of the cache, and said step of reading the first memory
7 block while reading the second memory block further includes
8 the steps of:

9 presenting the first memory block to the first
10 interconnect by connecting the first interconnect to an
11 output of the first multiplexer; and

12 presenting the second memory block to the second
13 interconnect by connecting the second interconnect to an
 output of the second multiplexer.

1 5

2 2

3 9. The method of Claim 8 wherein, if an error occurs
4 when examining a particular address tag as part of said step
5 of comparing the address tags stored in the first congruence
6 class, then a redundant address tag is substituted for the
7 particular address tag by examining a line of the second
 cache directory which corresponds with the line in the first
 cache directory containing the particular address tag.

ZG

M&A37

1 10. A computer system comprising:
2 a processor;
3 a memory device;
4 a system bus connected to said memory device;
5 a cache having a plurality of cache lines for storing
6 memory blocks corresponding to addresses of said memory
7 device; and
8 means for simultaneously reading a first memory block
9 from said cache and reading a second memory block from said
10 cache.

1 11. The computer system of Claim 10 wherein said
2 simultaneous reading means includes first and second cache
3 directories.

1 12. The computer system of Claim 10 wherein said
2 simultaneous reading means reads said first memory block and
3 said second memory block in a single clock cycle of said
4 processor.

1 13. The computer system of Claim 11 wherein said first
2 and second cache directories are redundant, and further
3 comprising means for writing an address tag of a memory
4 block which is stored in said cache to a specific line of
5 said first cache directory and to a specific line of said
6 second directory that corresponds to said specific line of
7 said first cache directory.

subarray
1 14. The computer system of Claim 11 wherein said cache
2 has a single cache entry array, and said simultaneous
3 reading means includes means for:

4 constructing a first control signal for said first
5 memory block based on a first location in said first
6 directory of an address tag associated with said first
7 memory block;

8 applying said first control signal to a first
9 multiplexer having inputs connected to said cache entry
10 array;

11 constructing a second control signal for said second
12 memory block based on a second location in said second
13 directory of an address tag associated with said second
14 memory block; and

15 applying said second control signal to a second
16 multiplexer having inputs connected to said cache entry
17 array.

1 15. The computer system of Claim 11 wherein said cache
2 has first and second cache entry arrays, and said
3 simultaneous reading means includes means for:

4 constructing a first control signal for said first
5 memory block based on a first location in said first
6 directory of an address tag associated with said first
7 memory block;

8 applying said first control signal to a first
9 multiplexer having inputs connected to said first cache
10 entry array;

11 constructing a second control signal for said second
12 memory block based on a second location in said second
13 directory of an address tag associated with said second
14 memory block; and

15 applying said second control signal to a second
16 multiplexer having inputs connected to said second cache
17 entry array.

1 16. The computer system of Claim 11 wherein each of
2 said first and second cache directories have a plurality of
3 congruence classes each having a plurality of lines for
4 storing said address tags, and said simultaneous reading
5 means further includes means for:

6 associating a first requested address with a first
7 congruence class in said first cache directory;

8 comparing each of said address tags stored in said
9 first congruence class with a portion of said first
10 requested address;

11 associating a second requested address with a second
12 congruence class in said second cache directory; and
13 comparing each of said address tags stored in said
14 second congruence class with a portion of said second
15 requested address.

1 12 9
2 17. The computer system of Claim 14 further comprising
3 a first interconnect for communicating with said processor,
4 and a second interconnect for communicating with said system
5 bus, and wherein said simultaneous reading means further
6 includes means for:

7 presenting said first memory block to said first
8 interconnect by connecting said first interconnect to an
9 output of said first multiplexer; and

10 presenting said second memory block to said second
11 interconnect by connecting said second interconnect to an
 output of said second multiplexer.

1 13 10
2 18. The computer system of Claim 15 further comprising
3 a first interconnect for communicating with said processor,
4 and a second interconnect for communicating with said system
5 bus, and wherein said simultaneous reading means further
6 includes means for:

7 presenting said first memory block to said first
8 interconnect by connecting said first interconnect to an
9 output of said first multiplexer; and

10 presenting said second memory block to said second
11 interconnect by connecting said second interconnect to an
 output of said second multiplexer.

30

1 *14*
2 19. The computer system of Claim 16 wherein, if an
3 error occurs when examining a particular address tag as part
4 of said comparing of said address tags stored in said first
5 congruence class, then a redundant address tag is
6 substituted for said particular address tag by examining a
7 line of said second cache directory which corresponds with a
8 line in said first cache directory containing said
 particular address tag.

00000000000000000000000000000000

31